

1. A method for managing communications between requester processes and server processes in a data processing network, including:

3 creating a set of dispatcher processes, each having a unique process identifier;

4 associating each of a set of requester processes, which communicate with a server process via a common interpreter process having a single process identifier, with a different dispatcher process of said set of dispatcher processes;

7 for requests sent from any of said set of requester processes via said common interpreter process to a server process which identifies requester processes using a process identifier, routing said requests via the associated dispatcher process;

10 at the respective dispatcher process, attaching the unique identifier of the dispatcher process to the request and then forwarding the request to the server process; and

12 responsive to receipt by the dispatcher process of a reply to said request, forwarding the reply to the associated requester process via the common interpreter process.

2. A method according to claim 1, wherein the common interpreter process via which said set of requester processes communicate is a Java Virtual Machine.

1 3. A method according to claim 2, wherein the set of requester processes comprise Web Browsers which communicate with a server process via respective Servlet threads running within a JVM of a Web Server or Web application server.

1 4. A computer program product comprising program code recorded on a machine readable
2 recording medium, the program code including instructions for, when executed, controlling the
3 operation of a data processing apparatus to implement a method for managing communications
4 between requester processes and server processes in a data processing network, the method
5 including:

6 creating a set of dispatcher processes, each having a unique process identifier;

7 associating each of a set of requester processes, which communicate with a server process

8 via a common interpreter process having a single process identifier, with a different dispatcher
9 process of said set of dispatcher processes;

10 for requests sent from any of said set of requester processes via said common interpreter
11 process to a server process which identifies requester processes using a process identifier, routing
12 said requests via the associated dispatcher process;

13 at the respective dispatcher process, attaching the unique identifier of the dispatcher
14 process to the request and then forwarding the request to the server process; and

15 responsive to receipt by the dispatcher process of a reply to said request, forwarding the
16 reply to the associated requester process.

1 5. A data processing apparatus, including:

2 a server process which uses process identifiers to distinguish between requests received
3 from different client processes;

4 means for creating a set of dispatcher processes, each having a unique process identifier;

5 means for associating each of a set of requester processes, which communicate with the

6 server process via a common interpreter process having a single process identifier, with a
7 different dispatcher process of said set of dispatcher processes;

8 means for routing requests from a requester process, comprising requests sent to the
9 server process from any of said set of requester processes via the common interpreter process, via
10 the respective associated dispatcher process;

11 means associated with the respective dispatcher process for attaching the unique
12 identifier of said respective dispatcher process to the request and then forwarding the request to
13 the server process; and

14 means responsive to receipt by said respective dispatcher process of a reply to said
15 request, for forwarding the reply to the associated requester process.